

MODUL 5
SOUND, ANIMATION, SHOWING TEXT, AND GAME ENDING



Adam Arthur Faizal

M3119001

TI A

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS SEBELAS MARET
SURAKARTA

2020

MODUL 5. Sound, Animation, Showing Text, Game Ending

Capaian Pembelajaran Praktikum:

- Memahami teknik animate objek
- Mendefinisikan method baru
- Menerapkan collision detection di Greenfoot
- Membuat game ending.

Tools:

- Java Development Kit (JDK)
- Greenfoot IDE

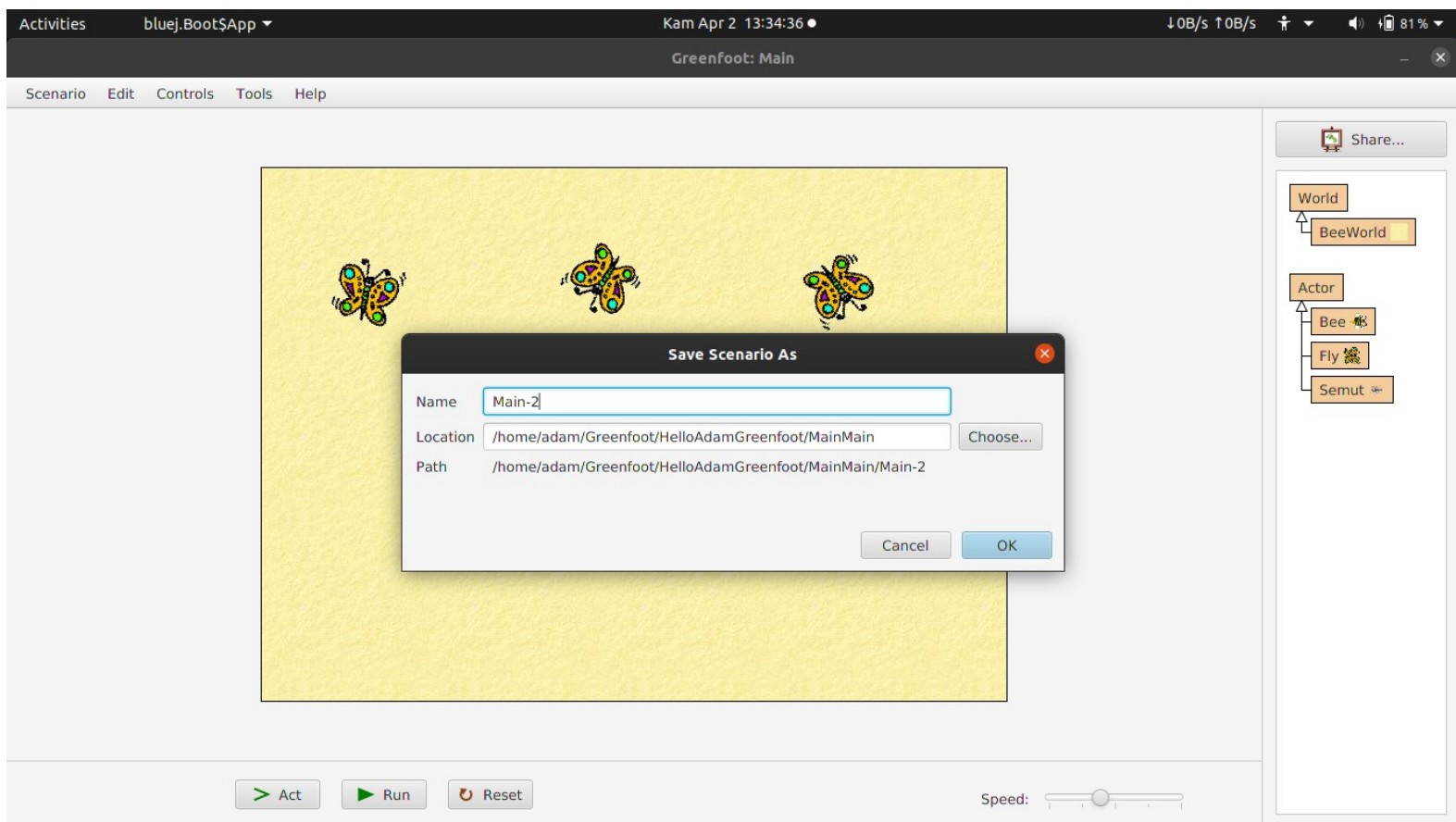
Terminologi:

Isikan terminology yang sesuai untuk definisi dibawah ini:

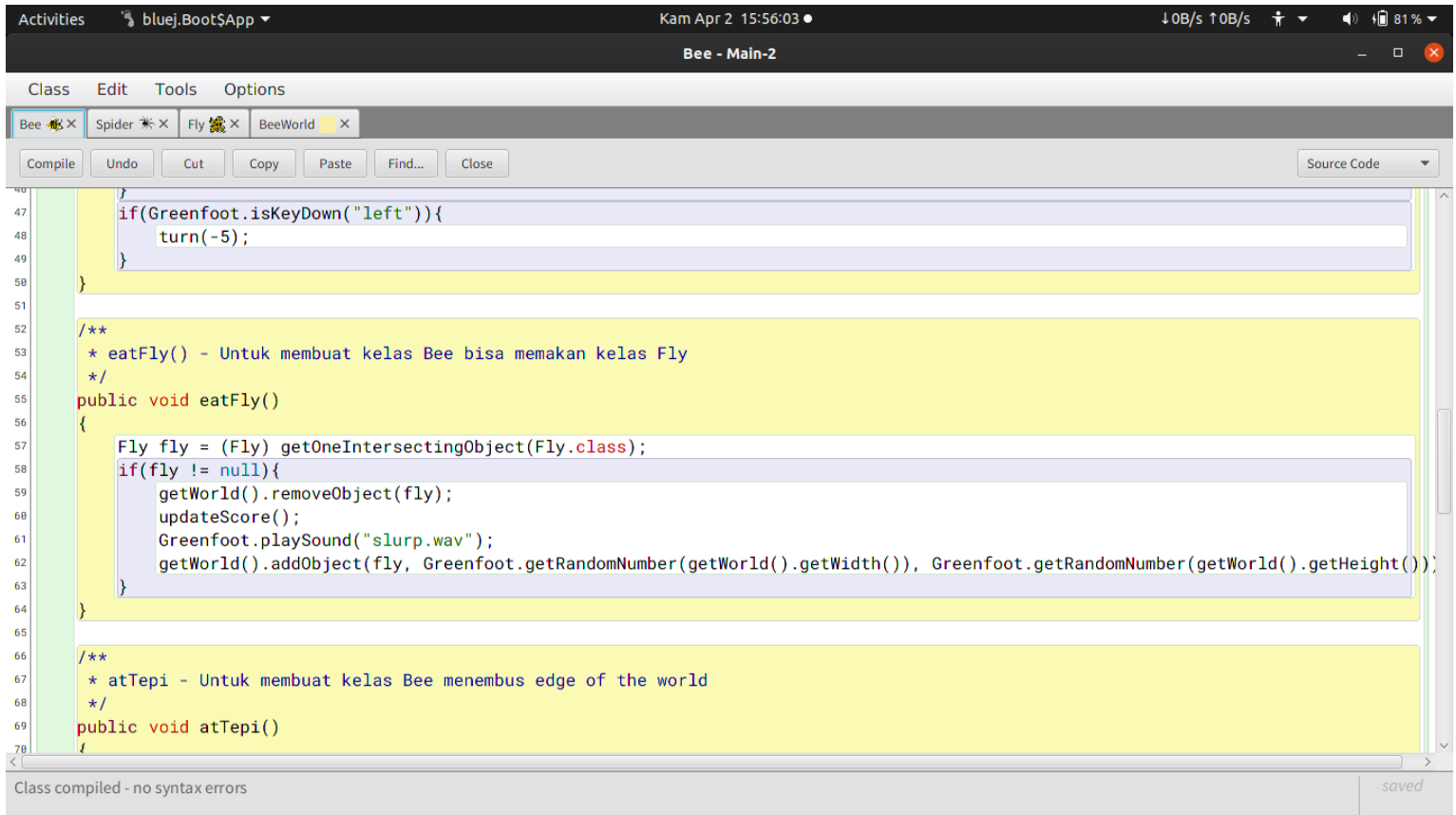
- | | |
|--------------------|--|
| [Pseudocode] | These statements describe the tasks or operations for the instances to perform in a mix of Java language and plain English words. This helps us better understand what behaviors we want the instances to perform before we write the real code. |
| [Defined Variable] | Declared in a class, this is used to store information for later use, or to pass information. It can store objects or values. |
| [Constructor] | A special method that is executed automatically whenever a new instance of the class is created. |

TRY IT / SOLVE IT:

1. Buka scenario modul sebelumnya, kemudian lakukan versioning dengan melakukan save as. Beri nama folder yang baru untuk menyimpan versi scenario yang baru.



2. Tambahkan suara "slurp.wav" ketika objek Bee memakan objek Fly. (pakai method *playSound()*)

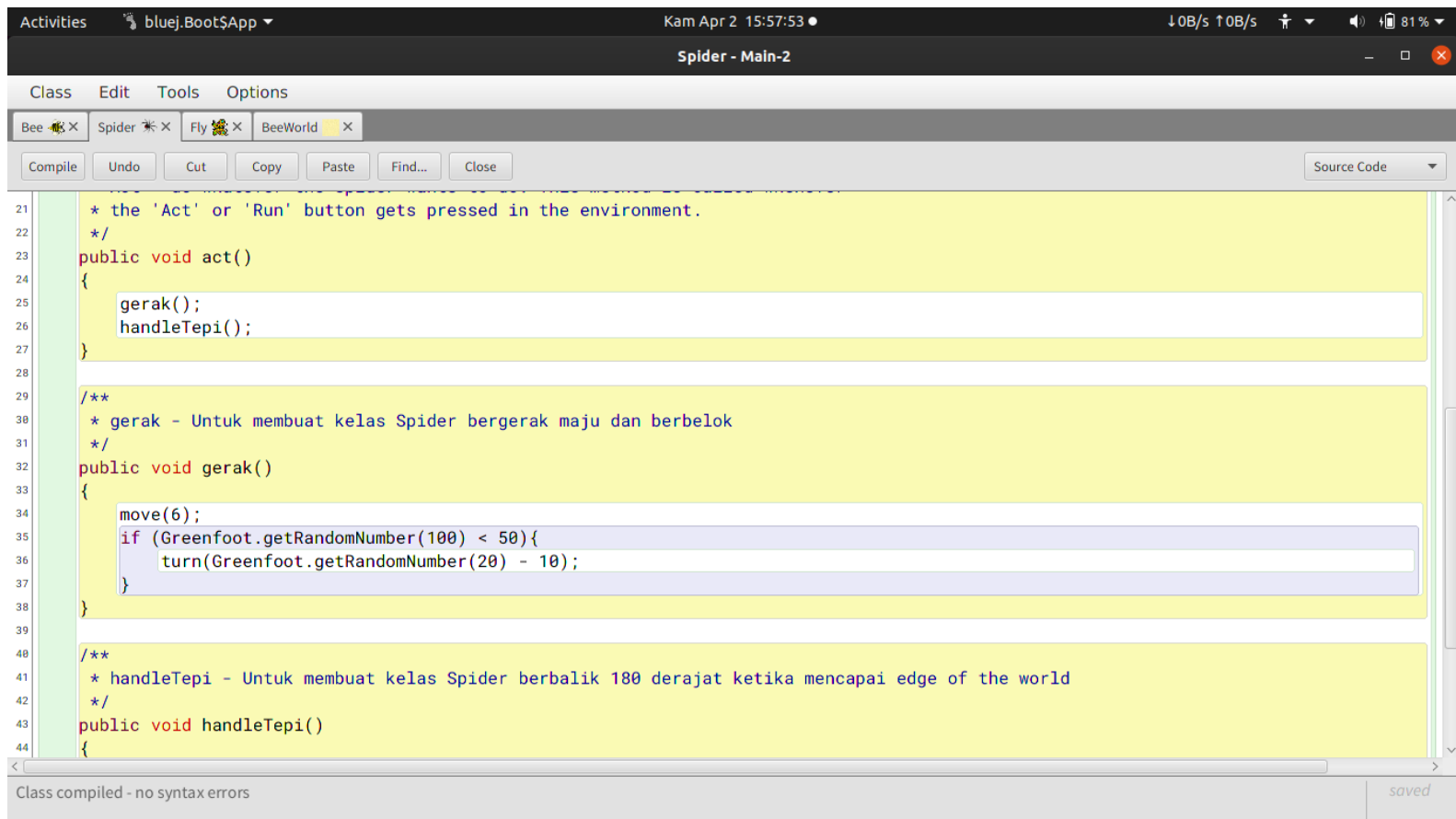


```
46 }
47 if(Greenfoot.isKeyDown("left")){
48     turn(-5);
49 }
50 }
51
52 /**
53  * eatFly() - Untuk membuat kelas Bee bisa memakan kelas Fly
54  */
55 public void eatFly()
56 {
57     Fly fly = (Fly) getOneIntersectingObject(Fly.class);
58     if(fly != null){
59         getWorld().removeObject(fly);
60         updateScore();
61         Greenfoot.playSound("slurp.wav");
62         getWorld().addObject(fly, Greenfoot.getRandomNumber(getWorld().getWidth()), Greenfoot.getRandomNumber(getWorld().getHeight()));
63     }
64 }
65
66 /**
67  * atTepi - Untuk membuat kelas Bee menembus edge of the world
68  */
69 public void atTepi()
70 {
```

Class compiled - no syntax errors

saved

3. Buat class Spider. Class Spider memiliki method *gerak()* untuk bergerak maju dengan kecepatan 2 dan berbelok ke kanan dan ke kiri sebesar 10 secara random dengan probabilitas 50:50. (Gunakan image spider.png)

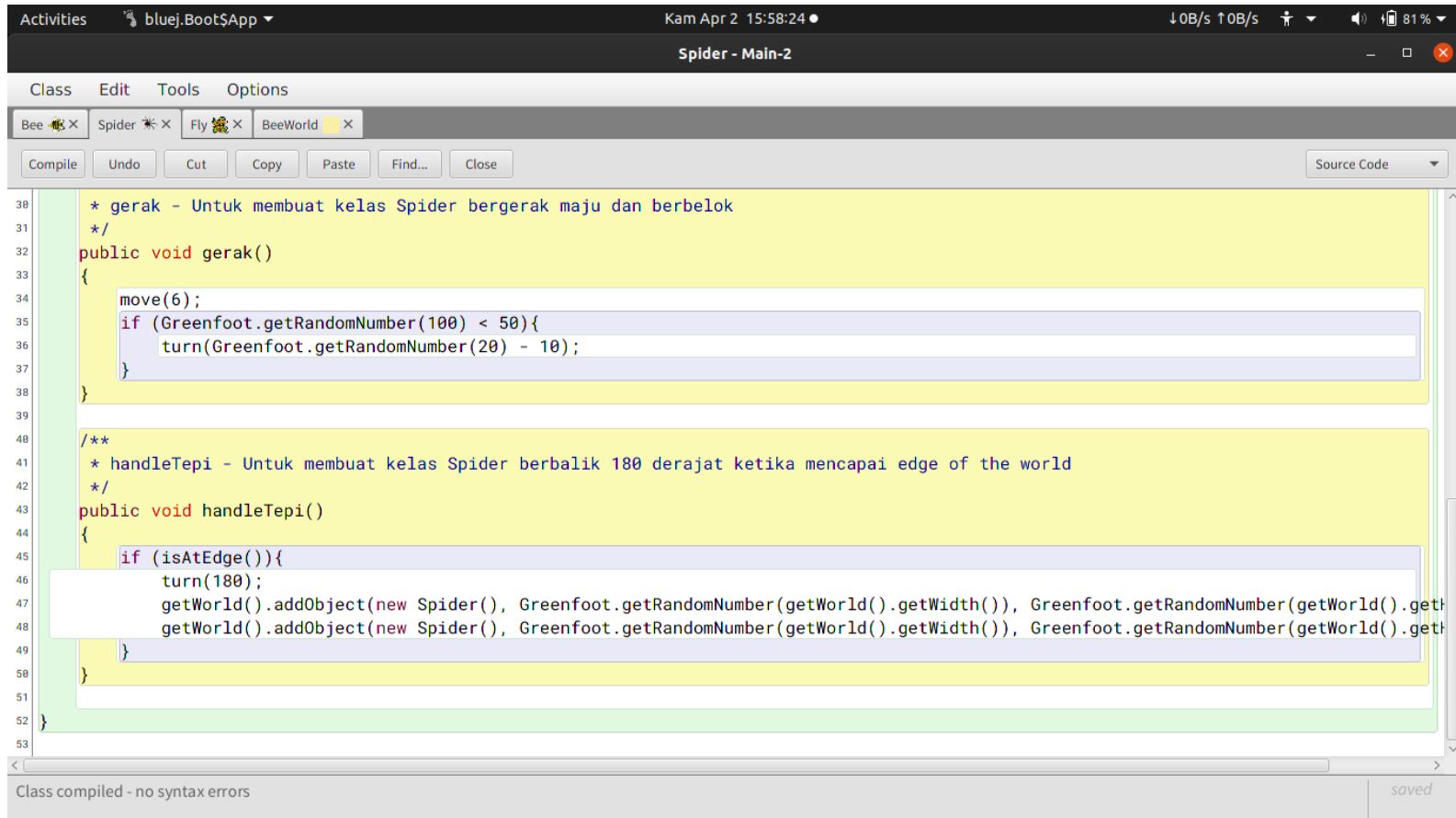


The screenshot shows a Java IDE window titled "Spider - Main-2". The code is as follows:

```
21  * the 'Act' or 'Run' button gets pressed in the environment.
22  */
23  public void act()
24  {
25      gerak();
26      handleTepi();
27  }
28
29  /**
30   * gerak - Untuk membuat kelas Spider bergerak maju dan berbelok
31   */
32  public void gerak()
33  {
34      move(6);
35      if (Greenfoot.getRandomNumber(100) < 50){
36          turn(Greenfoot.getRandomNumber(20) - 10);
37      }
38  }
39
40  /**
41   * handleTepi - Untuk membuat kelas Spider berbalik 180 derajat ketika mencapai edge of the world
42   */
43  public void handleTepi()
44  {
```

At the bottom of the IDE, a status bar indicates "Class compiled - no syntax errors" and a "saved" button is visible on the right.

4. Buat method *handleTepi()* yakni bilamana Spider mencapai tepi maka akan berbalik 180 derajat dan arah image nya juga berubah. Kemudian tambahkan dua objek Spider kedalam BeeWorld secara otomatis dengan posisi acak.

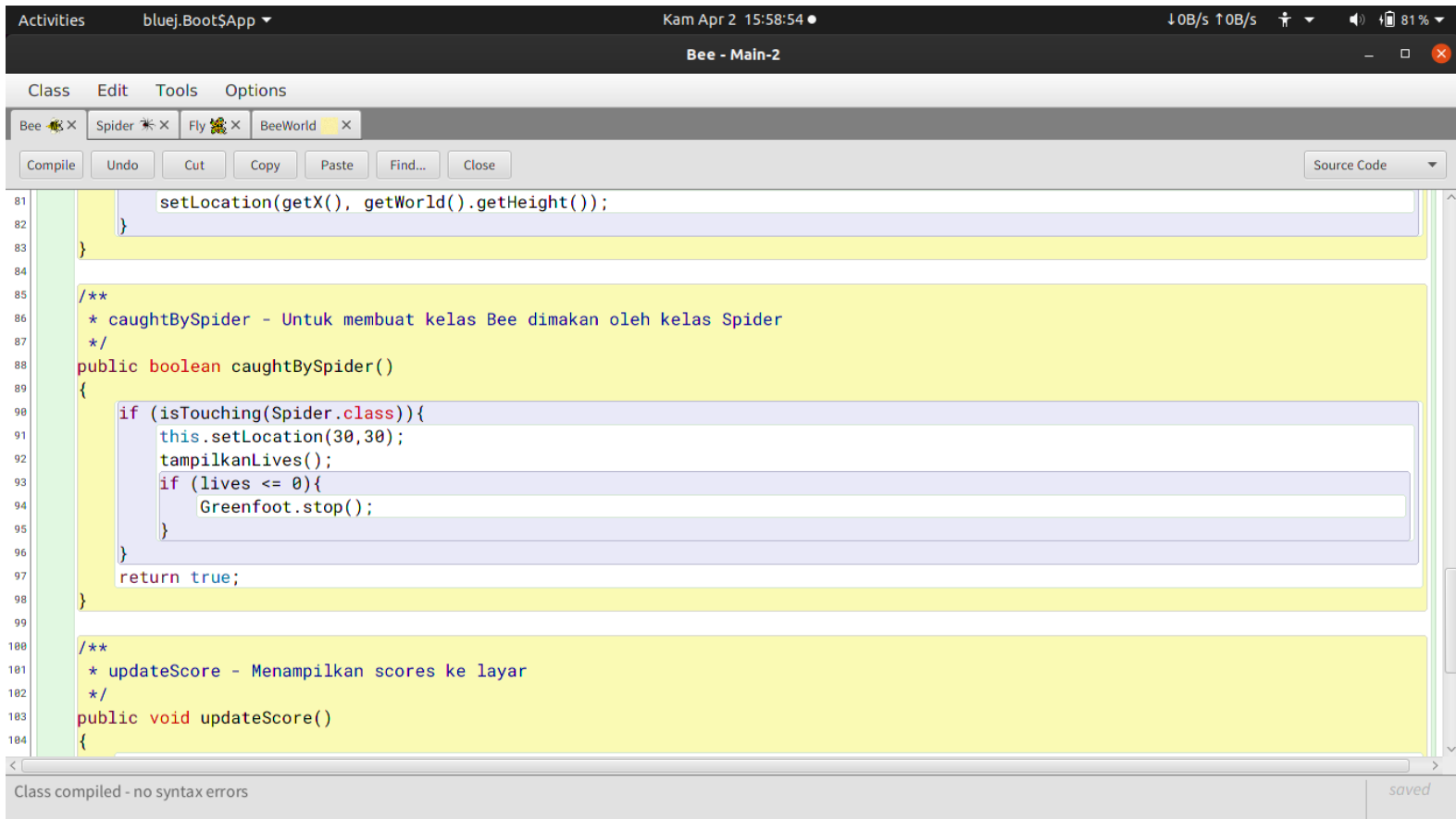


The screenshot shows a Java IDE window titled "Spider - Main-2". The code is for a class named `Spider`. It includes two methods: `gerak()` and `handleTepi()`. The `gerak()` method moves the spider 6 units and has a 50% chance to turn a random angle between -10 and 10 degrees. The `handleTepi()` method checks if the spider is at the edge of the world; if so, it turns 180 degrees and adds two new spider objects to the world at random positions.

```
30  * gerak - Untuk membuat kelas Spider bergerak maju dan berbelok
31  */
32  public void gerak()
33  {
34      move(6);
35      if (Greenfoot.getRandomNumber(100) < 50){
36          turn(Greenfoot.getRandomNumber(20) - 10);
37      }
38  }
39
40  /**
41   * handleTepi - Untuk membuat kelas Spider berbalik 180 derajat ketika mencapai edge of the world
42   */
43  public void handleTepi()
44  {
45      if (isAtEdge()){
46          turn(180);
47          getWorld().addObject(new Spider(), Greenfoot.getRandomNumber(getWorld().getWidth()), Greenfoot.getRandomNumber(getWorld().getHeight()));
48          getWorld().addObject(new Spider(), Greenfoot.getRandomNumber(getWorld().getWidth()), Greenfoot.getRandomNumber(getWorld().getHeight()));
49      }
50  }
51
52  }
53
```

Class compiled - no syntax errors

5. Tambahkan method *caughtBySpider()* di class Bee. Jika Bee bertemu dengan Spider, maka Bee seakan-akan dimakan oleh Spider dan berpindah ke lokasi awal dengan koordinat 30,30 dan lives/nyawa dari Bee berkurang satu satuan (lives awal dari Bee adalah 3). Ketika lives bernilai 0 atau kurang dari 0 maka permainan berhenti. (buat variabel untuk lives/nyawa)



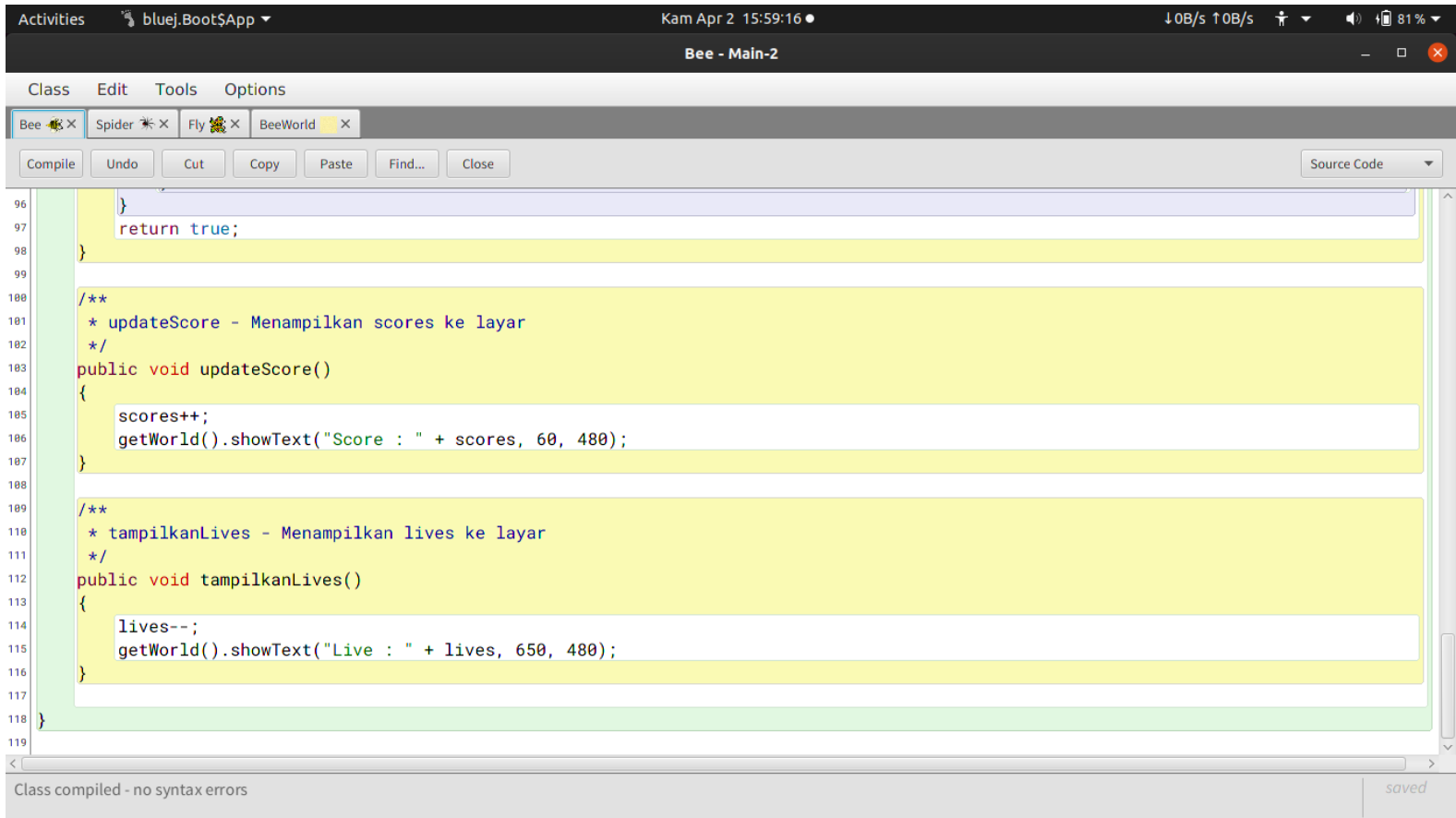
The screenshot shows a code editor window titled "Bee - Main-2". The editor has a menu bar with "Class", "Edit", "Tools", and "Options". Below the menu bar is a toolbar with buttons for "Compile", "Undo", "Cut", "Copy", "Paste", "Find...", and "Close". A "Source Code" dropdown menu is on the right. The code is written in Java and is as follows:

```
81     setLocation(getX(), getWorld().getHeight());
82 }
83
84
85 /**
86  * caughtBySpider - Untuk membuat kelas Bee dimakan oleh kelas Spider
87  */
88 public boolean caughtBySpider()
89 {
90     if (isTouching(Spider.class)){
91         this.setLocation(30,30);
92         tampilkanLives();
93         if (lives <= 0){
94             Greenfoot.stop();
95         }
96     }
97     return true;
98 }
99
100 /**
101  * updateScore - Menampilkan scores ke layar
102  */
103 public void updateScore()
104 {
```

At the bottom of the editor, a status bar indicates "Class compiled - no syntax errors" and "saved".

Tantangan:

6. Buatlah sebuah method *tampilkanLives()* di class Bee untuk menampilkan informasi mengenai jumlah lives yang tersisa di BeeWorld sehingga diketahui oleh player.



```
96     }
97     return true;
98 }
99
100 /**
101  * updateScore - Menampilkan scores ke layar
102  */
103 public void updateScore()
104 {
105     scores++;
106     getWorld().showText("Score : " + scores, 60, 480);
107 }
108
109 /**
110  * tampilkanLives - Menampilkan lives ke layar
111  */
112 public void tampilkanLives()
113 {
114     lives--;
115     getWorld().showText("Live : " + lives, 650, 480);
116 }
117
118 }
```

Class compiled - no syntax errors

Setelah sesi praktikum SELESAI, laporan praktikum dan source code (zip) harus dikirim/diupload ke google classroom sebelum praktikum minggu depan